

Amendments to the Claims:

A listing of the entire set of pending claims (including amendments to the claims, if any) is submitted herewith per 37 CFR 1.121. This listing of claims will replace all prior versions, and listings, of claims in the application.

Listing of Claims:

1. (Currently amended) Method of handling a group of at least one data object by issuing a data handling request to be processed by a storage device organised organized in allocation units by execution of at least one storage device request in a pre-determined data handling period, the method comprising the steps of:
 - a) determining the a number of data objects handling requests to be handled in the data handling period;
 - b) determining an upper boundary for the a number of allocation units involved for the per data handling request;
 - c) determining an upper boundary for the a number of storage device requests by multiplying the number of data handling requests as determined in step a) and the upper boundary of the number of allocation units involved;
 - d) determining an upper boundary for an amount of time consumed by execution of the data handling requests for handling the data objects during the data handling period by determining the an amount of time needed for execution of the upper bound for the number of storage device requests as determined in the previous step;
 - e) reserving on the amount of time as determined in the previous step in a data handling period for execution of the storage device requests; and
 - f) handling the data objects by executing the storage device requests.

2. (Currently amended) Method according to claim 1, wherein

- a) ~~the~~ maximum size of the data objects is substantially smaller than ~~the~~ size of allocation units;
- b) the data objects are stored non-contiguously at a substantially equal logic distance from each other such that multiple data objects can be stored in one allocation unit;
- c) the step of determining ~~an~~ of the upper boundary for the number of allocation units involved per data handling request is replaced by the step of determining an upper boundary of ~~the~~ number of data objects determined in step a) of claim 1 spaced at the substantially equal logic distance that is stored fragmented; and
- d) the step of determining an upper boundary for the number of storage device requests is replaced by the step of taking the sum of the number of data handling requests and the number of data objects determined in step c) of this claim.

3. (Currently amended) Method according to claim 1, wherein the method comprises the steps of including:

- a) determining ~~the~~ size of one allocation unit; and
 - b) determining ~~the~~ maximum size of a data object; and
- wherein the upper boundary for the number of allocation units involved is determined by the following expression:
- $$[\text{number of allocation units involved}] \leq ([\text{maximum size of a data object}] / [\text{size of one allocation unit}]) + 2,$$

4. (Currently amended) Method according to claim 2, wherein the method comprises the steps of including:

- a) determining ~~the~~ maximum distance between the data objects;
 - b) determining the size of one allocation unit; and
 - c) determining ~~the~~ maximum size of a data object; and
- wherein the upper boundary for the number of allocation units involved is determined by the following expression:

[number of storage device requests] ≤ [number of data handling requests] + ([number of allocation units involved] - 1)

wherein is and the upper boundary of the number of allocation units involved in execution of the data handling requests is determined by the following relation:
[number of allocation units involved] ≤ ([number of data handling requests] x ([maximum distance] + [maximum size]) / [size of allocation unit]) + 2.

5. (Currently amended) Method according to claim 1, wherein the data objects are include video frames comprised by a stream of audiovisual data.

6. (Currently amended) Method according to claim 5, wherein the stream of audio-visual data comprises-includes inter-coded and intra-coded frames.

7. (Currently amended) Method according to claim 6, wherein the multiple data objects to which the data handling requests are related are at least some of the intra-coded frames.

8. (Currently amended) Method according to claim 1, wherein step d) comprises the step of determining the upper boundary for the amount of time consumed by execution of the data handling request includes multiplying the upper boundary for the number of storage device requests by an amount of time consumed by a storage device request.

9. (Currently amended) Method according to claim 8, wherein the amount of time consumed by the storage device request is pre-determined.

10. (Currently amended) Method according to claim 1, wherein the storage device is a disk drive and the determination of the upper bound for the amount of time further takes into account at least one of the following parameters:

- a) ~~the an~~ amount of time required for one revolution of a disk;
- b) ~~the a~~ seek time of a pick-up unit of the disk drive to a location on a disk where a data object is located to which the data handling request is aimed; and
- c) ~~the a~~ time needed to retrieve the data object to which the data handling request is aimed.

11. (Currently amended) Method according to claim 2, wherein the storage device is a disk drive and the determination of the upper bound for the amount of time

consumed by execution of the data handling request further takes into account at least one of the following parameters:

- a) ~~the an~~ amount of time required for one revolution of a disk;
- b) ~~the a~~ seek time of a pick-up unit of the disk drive to a first location on a disk where a first data object is located to which the data handling request is aimed; and
- c) ~~the a~~ time needed to retrieve the data object to which the first data handling request is aimed; and
- d) ~~the a~~ time needed for the pick-up unit to move from the first location on the disk to a second location on the disk where a second, subsequent data object is located to which the data handling request is aimed.

12. (Currently amended) Method according to claim 1, wherein the determination of the upper boundary for the number of allocation units involved per data handling

requests comprises the step of includes dividing the size of the data object by the size of one allocation unit.

13. (Currently amended) Apparatus for handling a group of at least one data object by a data handling request to be processed by at least one storage device request handled in data handling periods, the data handling to be performed by a storage device organised organized in allocation units, the apparatus comprising a central processing unit conceived to:

- | a) determine the a number of data objects to be handled per data handling period;
- | b) determine an upper boundary for the a number of allocation units involved per data handling request;
- | c) determine an upper boundary for the a number of storage device requests by multiplying the number of data handling requests by the upper boundary of the number of allocation units involved;
- | d) determine an upper boundary for an amount of time consumed by execution of the storage device requests for handling the data objects during one data handling period by multiplying the upper boundary for the number of storage device requests by an amount of time consumed by a storage device request;
- | e) reserve an the amount of time consumed by execution of the storage device requests as determined in the previous step in a the data handling period for execution of the storage device requests; and
- | f) handle the data objects by executing the storage device requests.

14. (Currently amended) Computer programme product recorded on a computer readable medium that enables enabling a computer to be programmed to execute the method according to claim 1.

15. (Currently amended) Record carrier carrying the computer programme product according to claim 14.

16. (Original) Programmed computer enabled to execute the method according to claim 1.